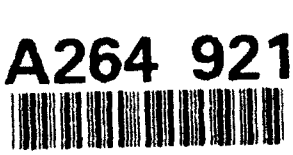

PL-TR-92-2131

FFTSCAN: A PROGRAM FOR SPECTRAL SMOOTHING USING FOURIER TRANSFORMS

W. O. Gallery
S. A. Clough

Atmospheric and Environmental Research, Inc.
840 Memorial Drive
Cambridge, MA 02139

19 May, 1992

Scientific Report No. 1

PHILLIPS LABORATORY
Directorate of Geophysics
AIR FORCE MATERIEL COMMAND
HANSCOM AIR FORCE BASE, MA 01731-3010

"This technical report has been reviewed and is approved for publication"

GAIL P. ANDERSON
Contract Manager

WILLIAM A.M. BLUMBERG, Chief
Simulation Branch

ROGER A. VAN TASSEL, Director
Optical Environment Division

This report has been reviewed by the ESC Public Affairs Office (PA) and is releasable to the National Technical Information Service (NTIS).

Qualified requestors may obtain additional copies from the Defense Technical Information Center. All others should apply to the National Technical Information Service.

If your address has changed, or if you wish to be removed from the mailing list, or if the addressee is no longer employed by your organization, please notify PL/TSI, Hanscom AFB MA 01731-3010. This will assist us in maintaining a current mailing list.

Do not return copies of this report unless contractual obligations or notices on a specific document requires that it be returned.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	19 May, 1992	Scientific #1

4. TITLE AND SUBTITLE
FFTSCAN: A Program for Spectral Smoothing Using Fourier Transforms

5. FUNDING NUMBERS
PE 62101F
PR 7670 TA 09 WU BG
Contract F19628-89-C-0150

6. AUTHORS
W. O. Gallery
S. A. Clough

7. PERFORMING ORGAINZATION NAMES(S) AND ADDRESS(ES)
Atmospheric and Environmental Research, Inc.
840 Memorial Drive
Cambridge, MA 02139

8. PERFORMING ORGANIZATION REPORT NUMBERS
(none)

9. SPONSORING/MONITORING AGENCY NAMES(S) AND ADDRESS(ES)
Phillips Laboratory
29 Randolph Road
Hanscom AFB, MA 01731-3010
Contract Manager: Gail Anderson/GPOS

10. SPONSORING/MONITORING AGENCY REPORT NUMBER
PL-TR-92-2131

11. SUPPLEMENTARY NOTES
Most of the work on this project by W. Gallery was performed under contract F19628-86-C-0182 while he was with OptiMetrics, Inc., Ann Arbor, Michigan.
The FFT routines supplied by Mark Esplin were formulated under AF Contract F19628-83-C-0056 with Stewart Radiance Laboratory.

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

12b. DISTRIBUTION CODE

13 ABSTRACT ***(Maximum 200 words)***

FFTSCAN is a computer program which uses Fourier transforms to convolve a spectrum with an instrument spectral scanning function. This technique mimics the operation of Fourier transform spectroscopy and preserves the full extent of the scanning function, which is particularly important for functions with large side lobes, like sin(x)/x. Five scanning functions commonly used in Fourier transform spectroscopy are available: sinc, $sinc^2$, Beer, Hamming, and Hanning, plus triangle and gauss. The program is specifically adapted to process spectra created by the atmospheric transmittance/radiance program FASCODE. A special, disk-based Fourier transform routine allows FFTSCAN to handle spectra of any size. FFTSCAN is more accurate and as or more efficient than the existing equivalent FASCODE routines.

This report describes the mathematical basis for spectral smoothing in the Fourier domain and gives the user instructions for FFTSCAN. It gives examples of spectral smoothing, including estimates of error limits and computational times, and provides guidance on implementing the program on different computers.

14 SUBJECT TERMS	15. NUMBER OF PAGES
Atmospheric Radiation, Spectral Smoothing, Fourier Transform Spectroscopy	24
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	SAR

Table of Contents

List of Figures

List of Tables

1. Introduction

A common application of FASCODE[1,2] calculations is a comparison to measured spectra. In order to make a proper comparison, the calculated spectrum must be convolved, or smoothed, with the spectral response function appropriate to the instrument being modeled. Frequently the measurements are made with a Fourier Transform Spectrometer (FTS) for which the spectral response function is known and which may be modified during data reduction[3]. By apodizing the interferogram, the shape and resolution of the spectral response function can be controlled.

In previous versions of FASCODE, spectral smoothing or "scanning" was performed by an actual convolution in the spectral domain. The scanning function is calculated over some finite spectral extent--typically some number of halfwidths or zeros from the line center--and is convolved with the calculated spectrum. The accuracy of the result depends strongly on the spectral extent: for a function like sinc, whose sidelobes fall off slowly, the scanning function must be calculated out to more than 100 halfwidths to achieve accuracies at the 0.2 percent level. However, the computational time varies linearly with the spectral extent of the scanning function so that accurate calculations with functions like sinc can be expensive.

We have added the capability to FASCODE to model the spectral response function associated with an FTS by using Fourier transforms. This technique directly mimics the operation of an FTS: the calculated spectrum is transformed into an "interferogram", "apodized", and then transformed back to the spectral domain. This technique is more accurate than the convolution technique and in many cases, more efficient. The user is given the choice of a number of commonly used apodization functions to define the shape of the scanning function, and may specify the resolution in terms of either the half width of the scanning function or the optical path difference of an equivalent interferometer.

---

[1] Clough, S., A., F. X. Kneizys, L. S. Rothman, and W. O. Gallery, 1981,Atmospheric spectral transmittance and radiance: FASCOD1B, Proc. of Soc. Photo. Opt. Instrm. Eng., **277**, 152-166.

[2] Clough, S. A., F. X. Kneizys, G. P. Anderson, E. P. Shettle, J. H. Chetwynd, L. W. Abreu, and L. A. Hall, 1989, FASCOD3: Spectral Simulation, in IRS '88: Current Problems in Atmospheric Radiation, J. Lenoble and J. F. Gelyn (Eds.), A. Deepak Pub.

[3] Bell, R. J., 1972, Introductory Fourier Transform Spectroscopy, Academic Press, New York

2. Theory

We will begin by briefly describing the mathematics of Fourier Transform Spectroscopy and then make the connection with the present work.

2.1 Fourier Transform Spectroscopy

The quantity measured by an interferometer is the interferogram $I(x)$ as a function of the optical path difference x. For an ideal spectrometer, I is related to the incident spectrum $S(\upsilon)$ through the Fourier transform $\mathcal{F}$:

$$I(x) = \mathcal{F}(S) = \int_{-\infty}^{+\infty} S(\upsilon) \exp(-2\pi i x \upsilon)\, d\upsilon \tag{1}$$

In practice, $I(x)$ is only measured out to L, the maximum optical path difference of the interferometer. The spectrum $S'(\upsilon)$ recovered from the interferogram is obtained from:

$$S'(\upsilon) = \mathcal{F}(AI) = \int_{-L}^{+L} A(x)\, I(x) \exp(-2\pi i x \upsilon)\, dx \tag{2}$$

where $A(x)$ is the apodization function applied to the interferogram to control the shape of the scanning function.

The convolution of two functions S and R is defined by the following expression:

$$S'(\upsilon) = R \star S = \int R(\upsilon)\, S(\upsilon'-\upsilon)\, d\upsilon' \tag{3}$$

where the symbol $\star$ represents convolution. A fundamental theorem of Fourier transforms states that the convolution of two functions equals the transform of the product of the transforms of the individual functions. Applying this theorem to Eq. 2 gives:

$$S'(\upsilon) = \mathcal{F}(A) \star \mathcal{F}(I) = R(\upsilon) \star S(\upsilon) \tag{4}$$

where $R(\upsilon) = \mathcal{F}(A(x))$. $R(\upsilon)$ is the scanning function associated with the apodization function $A(x)$.

If no apodization is applied, then $A(x)$ is effectively a rectangle of width equal twice the optical path difference. The scanning function associated with this apodization is the sinc function defined as:

$$\mathrm{sinc}(\upsilon) = \sin(2\pi i \upsilon L)/(2\pi i \upsilon L) \tag{5}$$

This function is characterized by relatively large side lobes which fall off slowly. It is possible to trade off resolution against smaller, more rapidly decaying side lobes by choosing different apodization functions. Figure 1 shows five common apodization/scanning function pairs. These figures correspond to the HIS[4] interferometer with a maximum path length difference of 1.3735 cm. This instrument was chosen as a representative Fourier transform spectrometer.

In practice, interferograms are discretely sampled on a grid $x_i = i\Delta x$, i= 0 to N-1, where Δx is the sampling interval and N is the number of points. (Since the ideal interferogram is symmetric around zero, only the positive half need be retained.) The continuous Fourier transform in Eq. 2 is replaced with the discrete transform:

$$S'(\upsilon_j) = \mathcal{F}(AI) = \sum_{i=0}^{N-1} A(x_i)\, I(x_i)\, \exp(-2\pi i x_i \upsilon_j) \qquad (6)$$

The frequency grid is given by $\upsilon_j = j\,\Delta\upsilon$, j= 0 to N-1, where $\Delta\upsilon = 1/(2L)$. The Nyquest frequency, $\upsilon_{max} = (N-1)\Delta\upsilon = 1/(2\Delta x)$, is the highest frequency that can be properly sampled. If frequencies higher than this are present in the signal, then they will be aliased down to lower frequencies in the recovered spectrum and the spectrum will be distorted.

2.2 Spectral Smoothing

In order to compare a FASCODE spectrum to a measured spectrum, the calculated spectrum must be convolved with the scanning function appropriate to the instrument being modeled. This convolution has been implemented using Fourier transforms:

$$S'(\upsilon) = \mathcal{F}(\ \mathcal{F}(R) \cdot \mathcal{F}(S)\ ) = R \star S \qquad (7)$$

where S is the monochromatic spectrum, R is the scanning function, and S' is the smoothed spectrum. By analogy to Fourier transform spectroscopy, $\mathcal{F}(S)$ is the interferogram and $\mathcal{F}(R)$ is the apodization function.

Note, however, that the calculated spectrum is limited from υ_1 to υ_2. Before transforming the spectrum, it is shifted in frequency down to the range 0 to υ_2-υ_1. In the interferogram domain, we now have $\Delta x = 1/(2(\upsilon_2-\upsilon_1))$ and $L = 1/(2\Delta\upsilon)$.

However, a significant difference exists between smoothing a calculated spectrum and apodizing an interferogram, having to do with edge effects. As a consequence of using discrete Fourier transforms, the measured spectrum must be thought of as repeating infinitely in the positive and negative directions, with a period of $\upsilon_{max} = 1/\Delta x$. Near the edges of the spectrum, at 0 and υ_{max}, the wings of the scanning function encounter the repeated spectra.

---

[4] Revercomb, H. E., H. Buijs, H. B. Howell, D. D. LaPorte, W. L. Smith, and L. A. Sromovsky, 1988, "Radiometric Calibration of IR Fourier Transform Spectrometers: Solution to a Problem with the High-Resolution Interferometer Sounder," Appl. Opt. **27**, p3210

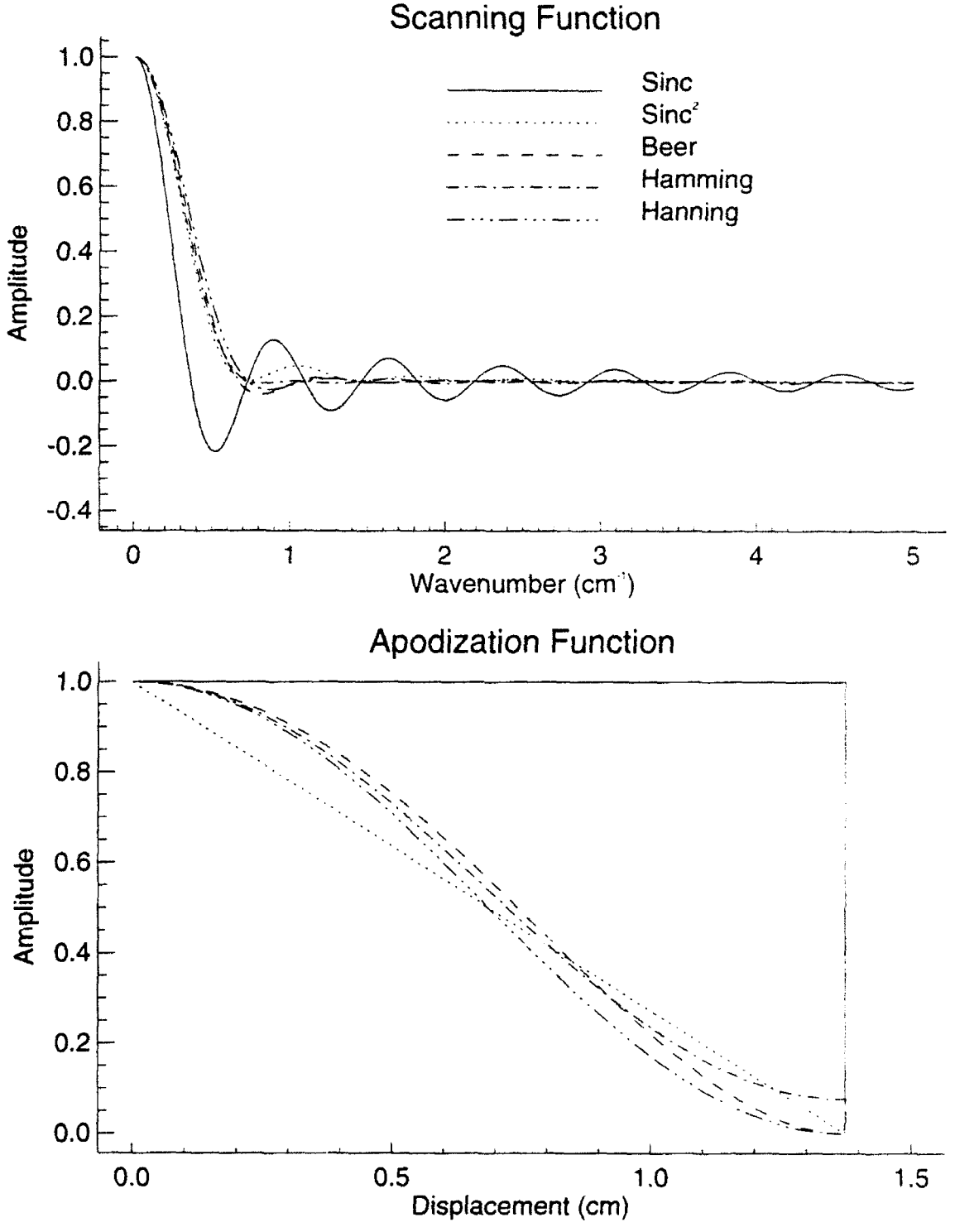

Figure 1. Five Common Scanning Function/Apodization Function Pairs: Sinc/Rectangle, $Sinc^2$/Triangle, $Beer/(1-(x/L)^2)^2$, Hamming, and Hanning. The maximum displacement of 1.3735 cm corresponds to the HIS instrument.

The calculated spectrum is limited from υ_1 to υ_2, or, after shifting, from 0 to υ_1-υ_2. This spectrum repeats, with a period of υ_2-υ_1, so that effectively υ_2 wraps around to υ_1. Near the edges of this spectrum, the wings of the scanning function encounter a spectrum different from that seen in the measured spectrum. This effect is especially severe when modeling unapodized spectra with the sinc scanning function, whose side lobes are particularly large. In order to minimize this problem, the range of the calculated spectrum (υ_1 to υ_2) should be made significantly larger than the actual range of interest.

The seven scanning functions included in FFTSCAN are listed in Table 1 along with the corresponding apodization function. For an FTS, the most convenient measure of the width of the scanning function is in terms of the parameter a which equals the reciprocal of the maximum optical path difference L of an equivalent interferometer. a is approximately the resolution of the apodized spectra, according to the Rayleigh criterion. It is also common to characterize the resolution of an instrument in terms of the halfwidth at half maximum (**HWHM**) and the distance to the first zero (**FZ**). The ratios a/HWHM and a/FZ are listed in Table 1 and can be used to convert from one form to another.

Theoretically, the "apodization" function $\mathcal{F}(R)$ can be computed either as the Fourier transform of the scanning function or analytically from the function listed in Table 1. The analytic method is more efficient but it assumes that the scanning function is infinite in extent. Because discrete transforms are used, the extent of the scanning function is limited to $\upsilon_2 - \upsilon_1$. The discrete transform of the scanning function over this extent will not exactly equal the apodization function, and if the halfwidth of the scanning function approaches the range $\upsilon_2 - \upsilon_1$, then the difference can become significant. This effect is illustrated in Figure 2. Here the scanning function is a sinc^2 with a HWHM of 0.3227 cm^{-1}, which corresponds to the resolution of the HIS instrument. This scanning function is to be applied to a spectrum with an extent of 12.9 cm^{-1}, which corresponds to 40 halfwidths (HWHM/($\upsilon_2 - \upsilon_1$) = C_R in Table 1). The analytic apodization function is a triangle of base 1.376 cm. The plot labeled "Error in Scan Function" shows the difference between the analytic sinc^2 function and the FFT of the triangular apodization function. Similarly, the plot labeled "Error in the Apodization Function" shows the difference between the analytic triangle function and the FFT of the sinc^2 scanning function. This effect is greatest for functions with sharp edges, e.g. the rectangle and triangle. The column C_R in Table1 lists for each function the critical ratio of ($\upsilon_2 - \upsilon_1$)/HWHM at which the maximum error in the scan function due to this effect becomes approximately 0.05 percent.

In view of this fact, the program has been designed with the capability to calculate the apodization function by either method. By default, the program chooses which method to use based upon the ratio ($\upsilon_2 - \upsilon_1$)/HWHM. This default can be overridden (see user instructions.)

Table 1. Scanning Function/Apodization Function Pairs

#	Name	Scanning Function	Apodization Function	*a*/HWHM	*a*/FZ	C_R
1.	Triangle/Sinc^2	$1 - \upsilon/a$, $\|\upsilon\| \| a$, 0: $\|\upsilon\| > a$	$(\sin(\pi x a)/(\pi x a))^2$	2.0	1.0	40
2.	Gauss/Gauss	$\exp(-0.5\ (\upsilon/a)^2)$	$\exp(-2\pi\ (a.c)^2)$	0.849322	(NA)	10
3.	Sinc^2/Triangle	$(\sin(\pi\upsilon/a)/(\pi\upsilon/a))^2$	$1 - xa$, $\|x\| \le 1/a$, 0: $\|x\| > 1/a$	2.257609	1.0	40
4.	Sinc/Rectangle	$\sin(u)/(u)$	1: $\|x\| \le 1/a$, 0: $\|x\| > 1/a$	3.314800	2.0	160
5.	Beer	$J(5/2,u)/u^{(5/2)}$	$(1 - (xa)^2)^2$	2.100669	91728	20
6.	Hamming	$\text{sinc}(u) + c_1\ (\text{sinc}(u+\pi) + \text{sinc}(u-\pi))$	$(1 + 2c_1 \cos(\pi x a))/(1+2c_1)$	2.195676	1.0	20
7.	Hanning	$\text{sinc}(u) + .5\ (\text{sinc}(u+\pi) + \text{sinc}(u-\pi))$	$(1 + \cos(\pi x a))/2$	2.0	1.0	20

Notes:

a. υ = frequency, in cm^{-1}, x = optical path difference, in cm, $u = 2\pi\upsilon/a$

b. $a = 1/L$, where L is the maximum optical path difference of an equivalent interferometer. a determines the resolution, or the width of the scanning function.

c. **HWHM** the halfwidth at half maximum of the scanning function.

d. **FZ** is the distance from the center of the scanning function to the first zero.

e. $\mathbf{C_R}$ = the critical value of the ratio of the extent of the spectrum $\upsilon_2 - \upsilon_1$ and the HWHM. When this ratio is less than $\mathbf{C_R}$, the apodization function is calculated as the FFT of the scanning function. When the ratio is greater, it is calculated analytically. (See the text.)

f. $J(5/2,u)/u^{(5/2)} = ((3-u^2)\sin(u) - 3u\cos(u))/u^5$; $J(n,u)$ is the Bessel function of order n

g. $c_1 = 0.428752$

By necessity, both the interferogram and the recovered spectrum from an FTS are real functions. The transform of a real function is symmetric and that of a symmetric function is real, so that both the interferogram and the spectrum must be symmetric. If only a one-sided interferogram is measured, then it must be reflected around the origin to produce a two-sided interferogram before transforming. However, the convolution in Eq. 7 applies to any real functions, symmetric or not. Therefore, the calculated spectrum S need not be reflected to produce a symmetric function before transforming. (The scanning function $R(\upsilon)$ is symmetric.) The "interferogram" $\mathcal{F}(S)$ will be a complex function but the convolved spectrum S' will again be real.

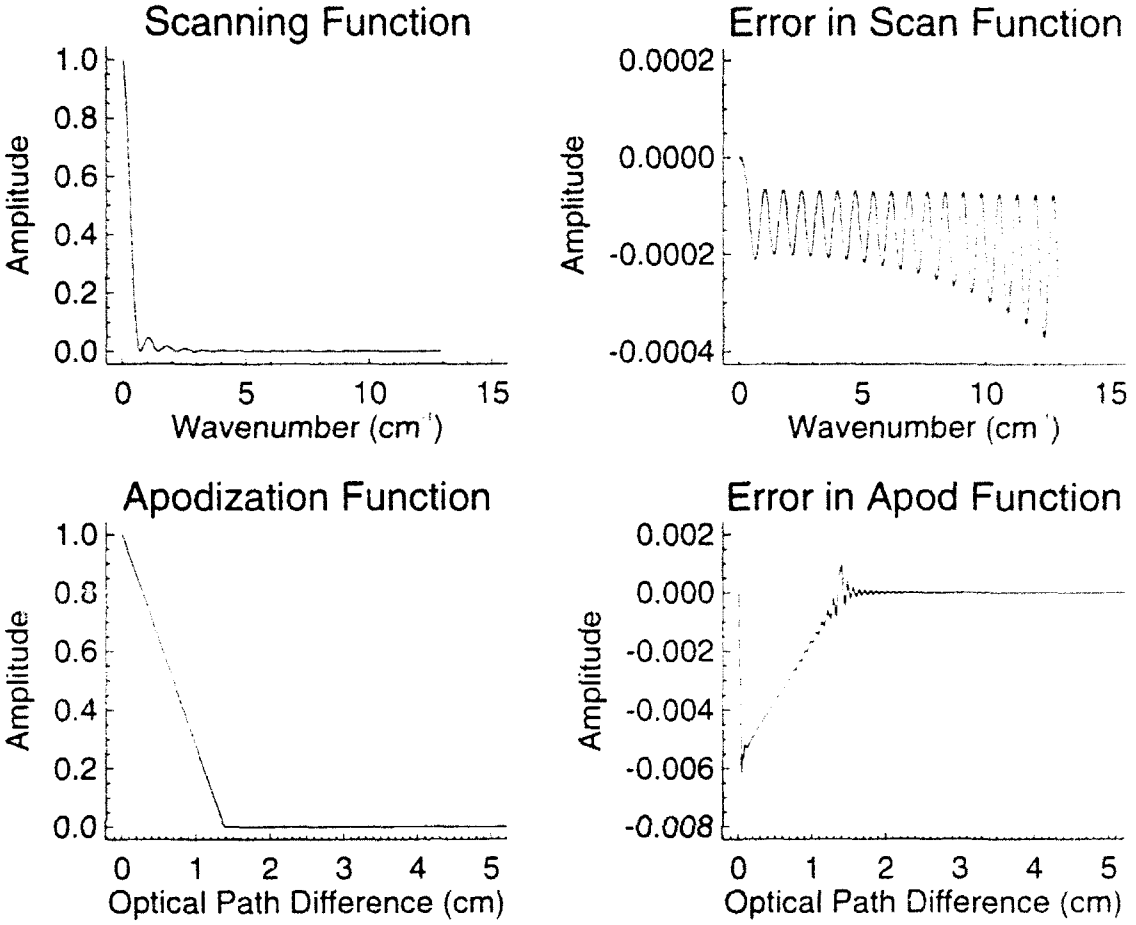

Figure 2. Analytic Versus Discrete Fourier Transform Calculations of: the $Sinc^2$ Scanning Function and the Triangular Apodization Function. The plot labeled "Error in Scan Function" is the difference between the $sinc^2$ function and the FFT of the triangle while the plot labeled "Error in Apod Function" is the difference between the triangle and the FFT of the $sinc^2$ function. See the text for further explaination.

2.3 Prescanning With a Boxcar

In cases where the width of the scanning function is large compared to the frequency spacing of the input spectrum, significant computational savings can be achieved by first convolving the spectrum with a rectangle whose width is small compared to that of the scanning function. In this procedure, M adjacent points are averaged to one output point at the mean frequency of the M points. This form of convolution, referred to here as prescanning with a boxcar, is very fast and reduces the number of output points by a factor of M. The resulting spectrum is then convolved with the desired scanning function using Fourier transforms.

Mathematically, the procedure relies on the fact that convolution is associative. Let S be the input spectrum, R_1 be a scanning function of halfwidth at half maximum α, and R_2 be a rectangle of half width M $\Delta\upsilon/2$. Then:

$$S'' = (S \star R_2) \star R_1 = S \star (R_2 \star R_1) \tag{8}$$

Let $r = \alpha/(M\ \Delta\upsilon/2)$. If $r \gg 1$, that is, if the rectangle R_2 is narrow compared to the scanning function R_1, then $R_2 \star R_1 \approx R_1$ and $S'' \approx S' = S \star R_1$. Here, "narrow" is taken to mean that $r \geq 8$. The error introduced by this approximation will be shown later.

It is also possible to partially compensate for the error introduced by boxcaring. This procedure is as follows. Using the notation from Eq. 4:

$$\begin{aligned} S' &= \mathcal{F}(S \star R_1) \\ &= \mathcal{F}(\ \mathcal{F}(S) \cdot \mathcal{F}(R_1)) \\ &= \mathcal{F}(\ \mathcal{F}(S) \cdot \mathcal{F}(R_2) \cdot \mathcal{F}(R_1) / \mathcal{F}(R_2)) \\ &= \mathcal{F}(\ \mathcal{F}(S \star R_2) \cdot \mathcal{F}(R_1) / \mathcal{F}(R_2)) \end{aligned} \tag{9}$$

In the last form of Eq. 9, dividing by $\mathcal{F}(R_2)$ removes the effect of the convolution of S with R_2. Since R_2 is a narrow rectangle, $\mathcal{F}(R_2)$ is a broad sinc function. In practice, the apodized "interferogram" is divided by $\mathcal{F}(R_2)$ before it is transformed back to the spectral domain.

The actual computation of S' is performed using discrete transforms, not integrals, and the convolution shown as $S \star R_2$ resamples the spectrum, taking only every M'th point. It is the resampling which produces the computational efficiency, since it reduces the number of points in the subsequent Fourier transforms by a factor of M. However, because of resampling, the deconvolution process is not exact; therefore, it is <u>not</u> possible to eliminate completely the smoothing effects of the boxcar. The error associated with boxcaring will be discussed next.

Boxcaring introduces two type of errors--frequency displacement and convolution error. The first type of error--frequency displacement--is illustrated in Figure 3 which shows the effect of boxcaring a delta function. As can be seen in the figure, the position of the delta function can be shifted by as much as M $\Delta\upsilon/2$, depending on the alignment of the resampled grid relative to the delta function. A similar shift can occur in a spectral feature of small but finite width, as will be seen later.

Boxcaring also introduces an error due to the convolution effect of the boxcar, which can only be partially compensated for by the deconvolution process. This error plus the frequency displacement error are both illustrated in Figure 4. In this example, the (synthetic) input spectrum is a single Lorenz line of HWHM = 0.08 cm^{-1} on a grid of 0.01 cm^{-1}. The scanning

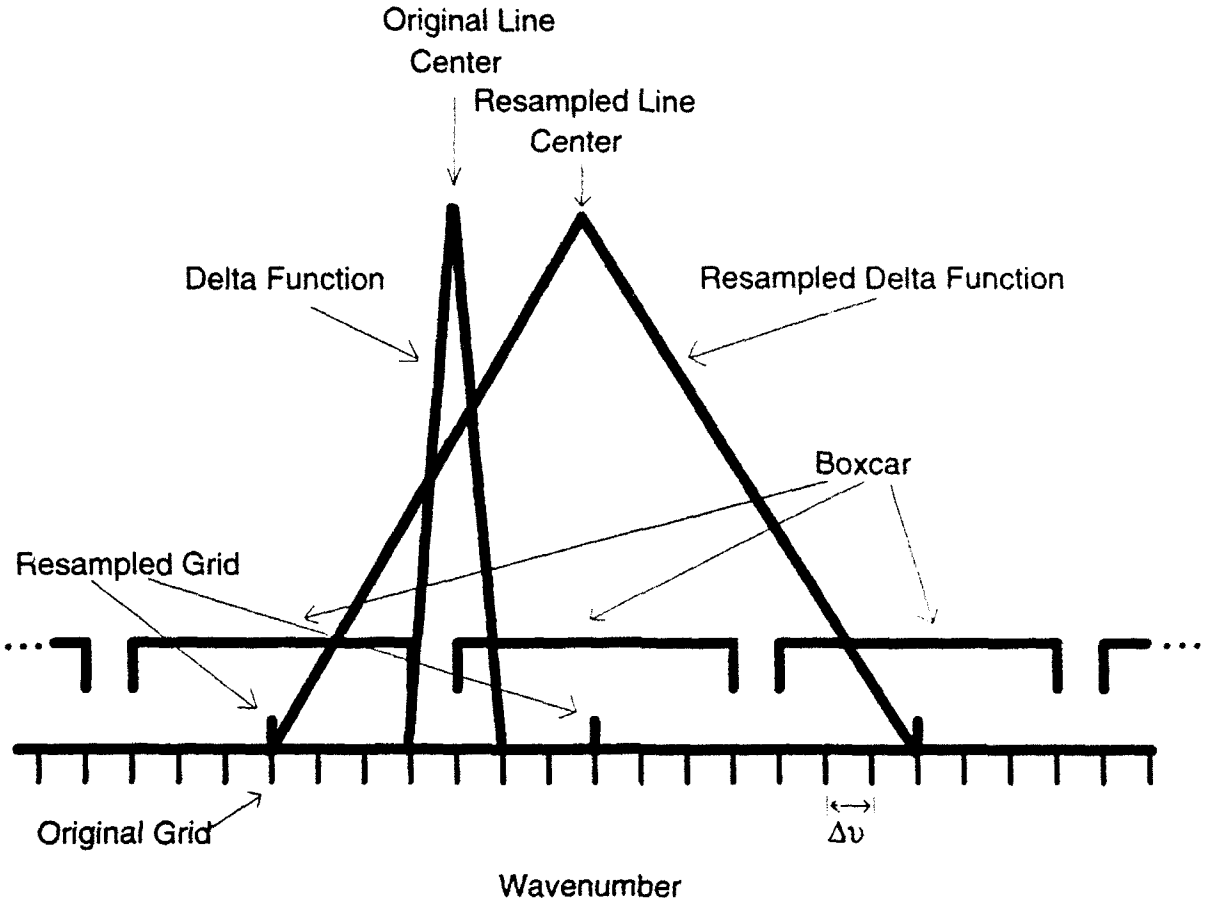

Figure 3. Schematic Diagram Showing the Effect of Boxcaring A Delta Function. The input spectrum containing a delta function is boxcared and resampled with a value of M = 7. The output spectrum contains a delta function shifted in frequency by $M\Delta\upsilon/2$.

function is a $sinc^2$ of HWHM = 0.3 cm^{-1}. A value of r = 8 gives a value of M = $2\times0.3/(8\times0.01)$ = 7 and a boxcar of half width = $7\times0.1/2$= .035 cm^{-1}. The output spectra are therefore on a grid of 0.07 cm^{-1}. The convolved spectrum has been calculated without boxcaring and with boxcaring for 4 different alignments of the resampled grid, each shifted by 0.01 cm^{-1}. The error spectra are the difference between the spectra with boxcaring and without boxcaring but resampled on the grid of the boxcared spectrum. The error spectra are shown both with and without deconvolution.

Considering the deconvolved spectra, the maximum error is as small as 0.05 percent, in the case where the line center falls on a resampled grid point (solid line), and as large as 0.2 percent when the line center falls halfway between the resampled grid points (dashed line). Without deconvolution, the corresponding errors are 0.2 and 0.3 percent respectively. Since the alignment of spectral features relative to the sampling grid is arbitrary, the maximum error in

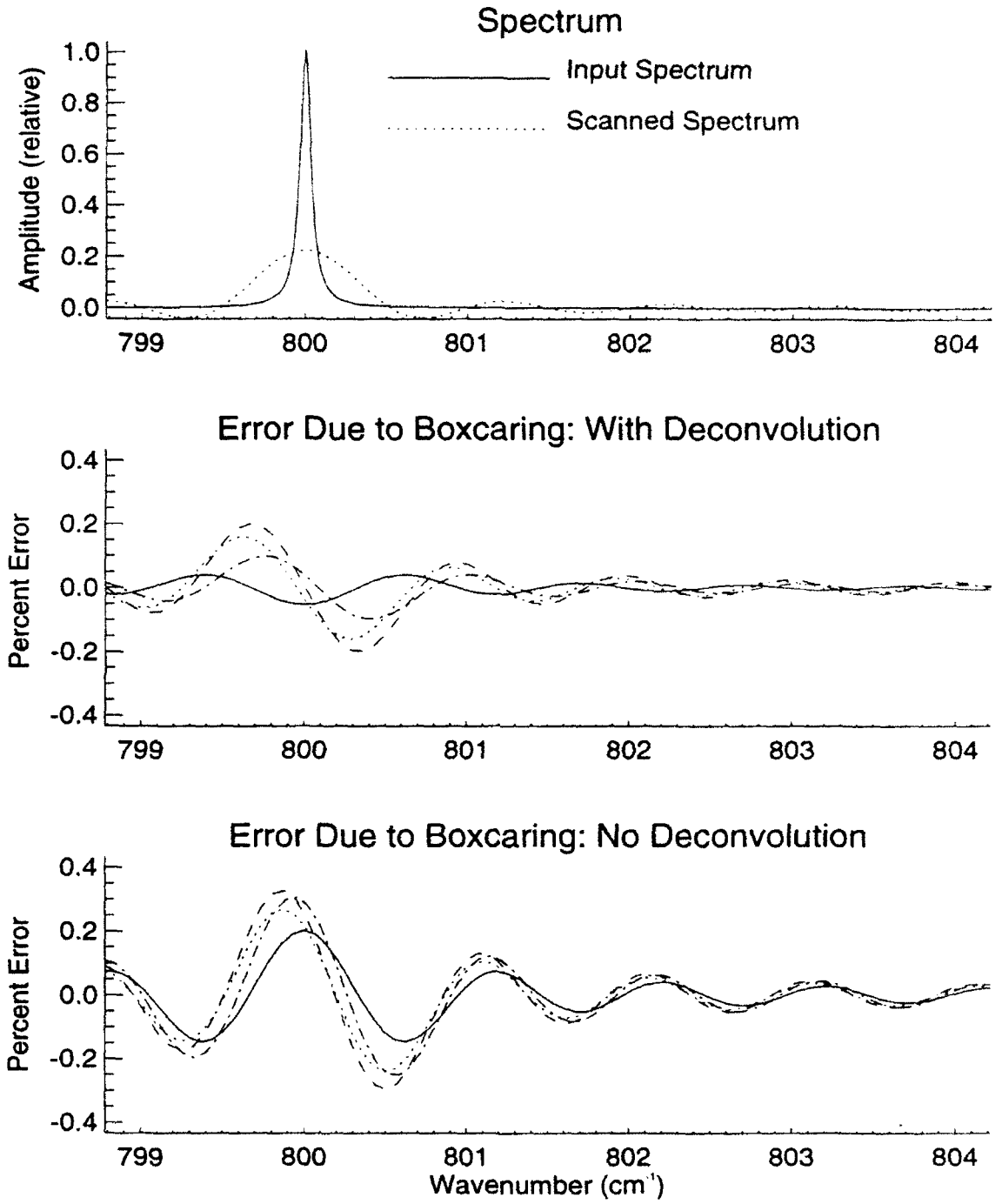

Figure 4. An Example of the Error, With and Without Deconvolution, Introduced by Boxcaring the Input Spectrum. The input spectrum consists of a single Lorenz line of HWHM of 0.04 cm^{-1} on a 0.01 cm^{-1} grid. The scanning function is a sinc of HWHM of 0.3 cm^{-1}, with $r = 8$ and M = 7, and the output grid spacing is 0.07 cm^{-1}. The four lines on each error plot correspond to different allignments of the output grid relative to the input grid (see text.)

each case must be assumed. Under this assumption, the deconvolution process decreases the boxcaring error by only one third. However, deconvolution represents only a small percent of the computational time so that deconvolution is still worth the computational cost.

The level of error seen in this figure is typical of the error for a value of MRATIO of 8. However, the actual error in a particular situation will vary from this example. For cases where high accuracy (error less than 0.5 percent) is required, the user is urged to experiment with different values of the parameters MRATIO and IVX and verify the accuracy for the particular application. The default value of MRATIO adopted for FFTSCAN is 12.

3. Fast Fourier Transform

The basic Fast Fourier Transform package used here is from Press[5]. Since the spectrum is a real function, its transform is Hermitian, that is, the negative components are the complex conjugate of the positive components and only the positive components need be stored. We used the subroutine `REALFT.FOR`, which can calculate both the forward transform of a real function to produce an Hermitian function, and the inverse transform of an Hermitian function to produce a real function.

A FASCODE spectrum can contain many thousands of points, more than can be stored in memory and transformed in place. Mark Esplin, of Stewart Radiance Lab has kindly provided a disk-based FFT routine which can transform an array of any size[6]. In the disk swapping FFT, the set of input data points is divided into blocks and written to a direct access file. Only two of these blocks of data reside in the central memory of the computer at a given time. As the transformation progress, these blocks of data are read, processed, and then rewritten to the same locations. As the routine proceeds, the data is first sorted into a particular order, an in-memory FFT (the same `REALFT.FOR` mentioned previously) is then applied to each block of data, and finally the data from the various blocks are combined to form the Fourier transformation of the entire data set. The manner in which the data are sorted into the appropriate blocks and the way the data from the blocks are combined into the Fourier transformation of the entire data set is analogous to that of the standard FFT. In this analogy, blocks of data correspond to the individual elements of the standard FFT and an array of data blocks on the mass storage device corresponds to the linear array of input data points. In addition to the blocks of data, there are disk blocks generated that contain sine-cosine information. The number of these blocks is 1/8 the number of data blocks. Both the size of each block and the number of data blocks need be a power of 2.

The parameter LPTSMX determines the block size for the FFT. (See Section 6 for a discussion of this parameter.) If the spectrum has fewer than LPTSMX points, then the FFT is done in memory: if it has more, the disk-based FFT is used. In either case, the spectrum is zero filled as needed so that the total number of points (and of blocks for the disk-based FFT) is a power of 2. The difference in computational time for the in-memory versus disk-based FFT for the same number of points varies depending on the target computer. The following table gives one example of a spectrum of $131072 = 2^{17}$ points scanned both ways on a Sun SparcStation. In this case, the disk-based case took almost 3 times as long as the in-memory case. On other machines, the ratio is reported to be smaller.

	LPTSMX	Blocks	Execution Time (sec)
In Memory	131072	1	21.5
Disk-Based	4096	32	55.5

[5] W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling, 1987: *Numerical Recipes in FORTRAN*, Cambridge University Press, NY.

[6] Mark Esplin, Stewart Radiance Lab, Bedford, MA., private communication.

4. Program Instructions and Notes

4.1 User Instructions

FFTSCAN is controlled by a single input record, similar to that for the normal scanning function. The details of this record are as follows:

Field:	HWHM	V1	V2	JEMIT	JFN	MRAT	DVOUT
Column:	1-10	11-20	21-30	31-35	36-40	41-45	46-55

Field:	IUNIT	FIL	NFIL	JUNIT	IVX	NOFIX
Column:	56-60	61-65	66-70	71-75	76-78	79-80

Format(3F10.3,3I5,F10.3,4I5,I3,I2)

HWHM Half Width at Half Maximum of the scanning function, or if JFN < 0, the maximum optical path difference of an equivalent interferometer. If HWHM ≤ 0, then exit FFTSCAN.

V1 Initial wavenumber for the scanned result

V2 Final wavenumber for the scanned result

JEMIT = 0: convolve with transmittance
= 1: convolve with radiance

JFN Selects the Scanning Function (See Table 1)
= 0: Boxcar. Halfwidth is truncated to M dυ/2, where M is an integer and dυ is the grid spacing of the input spectrum
= 1: Triangle
= 2: Gauss
= 3: $Sinc^2$
= 4: Sinc
= 5: Beer
= 6: Hamming
= 7: Hanning
If JFN < 0, then HWHM is the maximum optical path difference of an equivalent interferometer, apodized to give the scanning function given by |JFN|.

MRAT For prescanning with a boxcar, the ratio of HWHM of the scanning function to the halfwidth of the boxcar, default = 12. If MRAT < 0, no boxcaring is performed (see Notes.)

DVOUT Output grid spacing (Not used, reserved for future use)

IUNIT Unit number of the file containing the spectrum to be scanned, default = 11 (see Notes)

IFIL Sequential number of the first FASCODE file on IUNIT to be scanned

NFIL Number of FASCODE files on IUNIT to be scanned, beginning with IFIL

JUNIT Unit number of the file containing the output spectrum, default = 12 (see Notes)

IVX = -1: Scanning function is calculated as the FFT of the Apodization function
= 0: Program decides how to calculate the scanning function, using C_R from Table 1.
= 1: Scanning function is calculated analytically

NOFIX For prescanning with a boxcar: if non-zero, then do not deconvolve the scanned spectrum with the boxcar

4.2 Notes

The program expands the spectral interval v_1 to v_2 to v_1 - C_R(JFN)*HWHM to v_2 + C_R(JFN)*HWHM before smoothing (see Table 1.) This expansion ensures that edge effect do not contaminate the endpoints of the scanned spectrum. If there is insufficient data for this expansion, then the maximum expansion possible is performed and an informative message is written. The output spectrum extends from v_1-2dυ to v_2+2dυ, where dυ is the output spacing. The extra points at either end allow for four-point interpolation of the spectrum at the endpoints.

The default value of MRAT should reduce the boxcaring error sufficiently for most applications (better than 0.2 percent). For greater accuracy, it is necessary to increase MRAT, or set it to -1 to turn off boxcaring.

The spectral input and output files are on units IUNIT and JUNIT, which default to 12 and 11 respectively. For input, if no file is open on unit = IUNIT then the program looks for a file named TAPExx, where xx = IUNIT (e.g. TAPE12.) If a file by this name does not exist, then an error results and then the program stops. For output, if a file is open on unit = JUNIT , then that file is rewound and used. If not, then the program looks for a file by named TAPExx . If that file does exist, then an error results and the program stops. If it does not exist, then a new file by that name is opened for output. If IUNIT or JUNIT are negative, then the program reads in the spectral file names (60 characters maximum, including the path) on the next record. If the named input file does not exist or if the named output file does exist, then an error results and the program stops. Otherwise, they are opened on the first free unit numbers between 61 and 99.

5. Examples

Figure 5 shows an example of a FASCODE spectrum smoothed using FFTSCAN. The calculated spectrum models the upward radiance at 72 km for the US Standard Atmosphere. The monochromatic calculation extended from 600 to 800 cm^{-1} with a dυ of 0.000953. In Figure 5a, the monochromatic FASCODE spectrum was smoothed with a sinc scanning function of HWHM = 0.2196 cm^{-1}, corresponding to the HIS resolution. Figure 5b shows the error in the scanned spectrum from using boxcaring with deconvolution (r = 12). This value of r (or MRAT in the user instructions) gives a value of M of 38, resulting in a 38 fold reduction in the number of points in the spectrum. The maximum error of about $1x10^{-8}$ is about 0.5 percent of the typical spectral excursion of $2x10^{-6}$ or about 0.2 percent of the maximum spectral value of $8x10^{-8}$. For reference, Figure 5c shows the error using the standard FASCODE 3 convolution with a bound of 80 halfwidths. The reference spectrum for calculating the errors in Figures 5b and c is the FFTSCAN calculation without boxcaring, interpolated to the proper grid.

Table 2 compares the computational time and maximum error for the calculations shown in Figure 5. The calculations were performed on an Sun SparcStation and the scanning functions were applied from 625 cm^{-1} to 775 cm^{-1}. The results for scanning with the $sinc^2$ and the triangular scanning functions are also shown. Note that the boxcaring errors for the other functions are four times less than that for the sinc function.

These results show that for the sinc function, FFTSCAN with boxcaring provides a three-fold increase in speed and a better than two fold increase in accuracy over FASCODE. For the $sinc^2$ and the triangle, the execution time for the two programs is about equal, but FFTSCAN is again twice as accurate. The reference spectrum for calculating the errors is again the FFTSCAN calculation without boxcaring, interpolated to the proper grid.

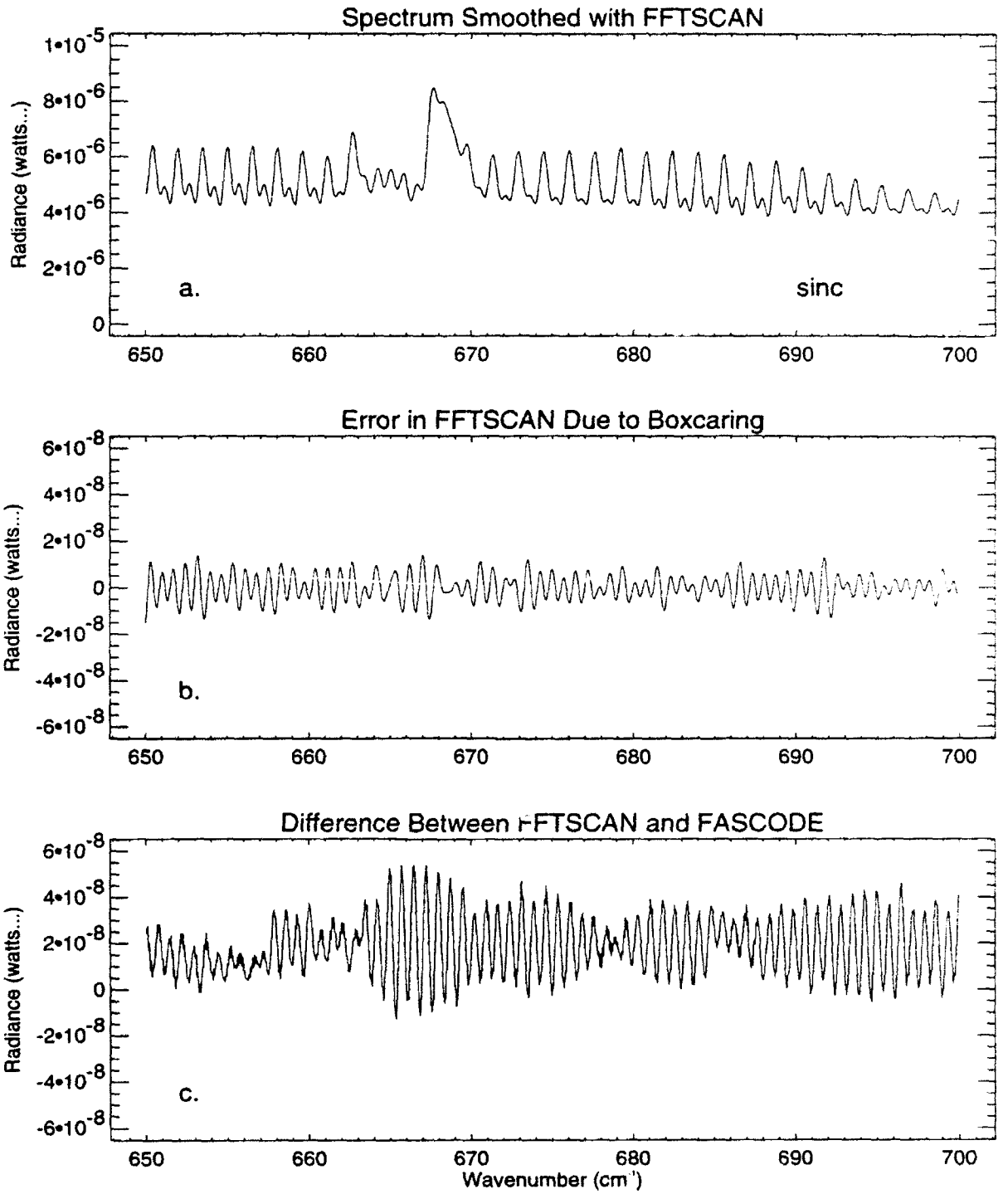

Figure 5. Example of a Smoothed Spectrum and the Associated Error: a. FASCODE Calculated Spectrum Smoothed by FFTSCAN, b. The Error In the Smoothed Spectrum from Using FFTSCAN With Boxcaring, and c. Error From Using the FASCODE3 Scanning Function. The scanning function is a sinc with a HWHH of 0.2196 cm^{-1}, corresponding to the HIS instrument in the unapodized mode.

Table 2. Comparison of Computational Time and Accuracy, FFTSCAN versus FASCODE, For the Sinc, Sinc^2, and Triangle Scanning Functions.

	Computational Time (sec)			Maximum Error (Percent Radiance)		
Scanning Option	**Sinc**	**Sinc^2**	**Triangle**	**Sinc**	**Sinc^2**	**Triangle**
FFTSCAN, No Boxcar	117	117	117	(NA)	(NA)	(NA)
FFTSCAN, With Boxcar	5.8	3.7	3.3	0.2	0.05	0.05
FASCODE3 Convolution	17.9	3.9	3.0	0.5	0.1	0.1

Notes:

a. The computational times refer to a Sun SparcStation. Times are approximate and both the absolute and the relative times will vary depending on the case.

b. The spectral extent of the smoothed spectrum was from 650 to 775 cm^{-1}

c. The monochromatic $d\upsilon$ was 0.000953 cm^{-1}

d. The number of points in the scanned function was 167,958 (noboxcaring) and 4419 (with boxcaring), a reduction of a factor of 38.

6. Implementation Notes

FFTSCAN is written in ANSI Standard Fortran 77 and is designed to be highly portable. It was developed on Sun workstation under Unix, and early versions have been ported to a VAX, a Cyber computer under NOS/VE and an IBM PC. FFTSCAN is designed either to be run as an independent program or to be included as a module of FASCODE.

The program uses three include files: `fftparm.inc`, `parmcomm.inc`, and `scancomm.inc`. These files carry parameters, common blocks, and other declarations used throughout the program. The file `scancomm.inc` includes the DOUBLE PRECISION statements:

```
Implicit Double Precision (V)
Double Precision XID,SECANT,HMOLID,XALTZ,YID
```

which may or may not have to be disabled, depending on how it is set in FASCODE. Typically, it is enabled on a 32 bit machine and disabled on a 64 bit machine. There is no standard syntax for the `include` statement and these statements may have to be changed to suit a particular system. For versions of Fortran which do not support `include` files, the program may be distributed with these files already included.

There are a few hardware dependencies related to the disk-based FFT which must be considered. If the size of the spectrum is greater than memory set aside for the in-memory FFT, then the program uses the disk-based FFT. The disk-based FFT divides the input data points into blocks and writes these blocks to disk as direct access records. An in-memory FFT is performed on each record. The size of each record must be a power of 2. The parameter LPTSMX sets the maximum size of an in-memory FFT and the size of the direct access records. This variable should be set as large as possible for the particular computer since the in-memory FFT is more efficient. The pitfall that must be avoided is setting LPTSMX too large in which case the virtual memory system will page the data. Since the points processed by the in-memory FFT come from widely scattered locations, the number of calls to the disk will be extremely large (thrashing) and the performance will be very poor. On the other hand, the minimum size of an FFT is also LPTSMX (this is a design error which will be corrected in the next version.) If LPTSMX is large but the region to be scanned is small, then the calculation will take unnecessarily long. Therefore, LPTSMX should be set to somewhere between the smallest typical spectral size and the largest value possible without thrashing. The parameter LPTSMX is set in the `include` file `fftparm.inc`. The user may have to adjust LPTSMX from the default value ($8192 = 2^{13}$) to be optimum for the target computer.

To determine whether the program is using the in-memory or the disk-based FFT, note the following line in the output:

```
FFT: Total number of points =     xxxxx  and blocks =      yy
```

If the number of points yy is 2 or greater, then the disk-based FFT is being used.

The parameter IBLKSZ, declared in the same statement, is the block size of the direct access records and is used in OPEN statements. Depending on the computer, it may be in words (e.g., VAX and CDC Cyber) in which case `IBLKSZ = LPTSMX`, or in bytes (Microsoft Fortran for the PC, Sun, and Alliant) in which case `IBLKSZ = 4*LPTSMX`. On a CRAY, which has a

word size of 64 bits (8 bytes), use `IBLKSZ = 8*LPTSMX`. In VAX Fortran, the maximum direct access block size is 4095 words so that the maximum allowed value of `IBLKSZ` is 2048. Since this is much smaller than the maximum physical memory, the user may want to use a large value of `LPTSMX` and disable the disk-based FFT capability.

On CDC Cyber machines, the subroutines `BUFIN` and `BUFOUT` may have to be changed to match the those used in `FASCODE`.

To modify `FFTSCAN` to include other scanning functions, the subroutines `FFTSCN.F` and `SCNFNT.F` need to be modified. In `FFTSCN.F`, modify the variables `JFNMAX`, `ANAMSES`, `C`, `CRATIO`, and `CLIMIT` as appropriate (the definitions of these variables are given in program comments.) In `SCNFNT.F`, use an existing scanning function as a example, and add the equations defining the new function, both in the frequency and in the space domain.

References

Clough, S., A., F. X. Kneizys, L. S. Rothman, and W. O. Gallery, 1981,Atmospheric spectral transmittance and radiance: FASCOD1B, Proc. of Soc. Photo. Opt. Instrm. Eng., **277**, 152-166.

Clough, S. A., F. X. Kneizys, G. P. Anderson, E. P. Shettle, J. H. Chetwynd, L. W. Abreu, and L. A. Hall, 1989, FASCOD3: Spectral Simulation, in IRS '88: Current Problems in Atmospheric Radiation, J. Lenoble and J. F. Gelyn (Eds.), A. Deepak Pub.

Bell, J. R., 1972, Introductory Fourier Transform Spectroscopy, Academic Press, New York

Esplin, Mark, Stewart Radiance Lab, Bedford, MA., private communication.

Press, W. H., B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling, 1987: *Numerical Recipes in FORTRAN*, Cambridge University Press, NY.

Revercomb, H. E., H. Buijs, H. B. Howell, D. D. LaPorte, W. L. Smith, and L. A. Sromovsky, 1988, "Radiometric Calibration of IR Fourier Transform Spectrometers: Solution to a Problem with the High-Resolution Interferometer Sounder," Appl. Opt. **27**, p3210